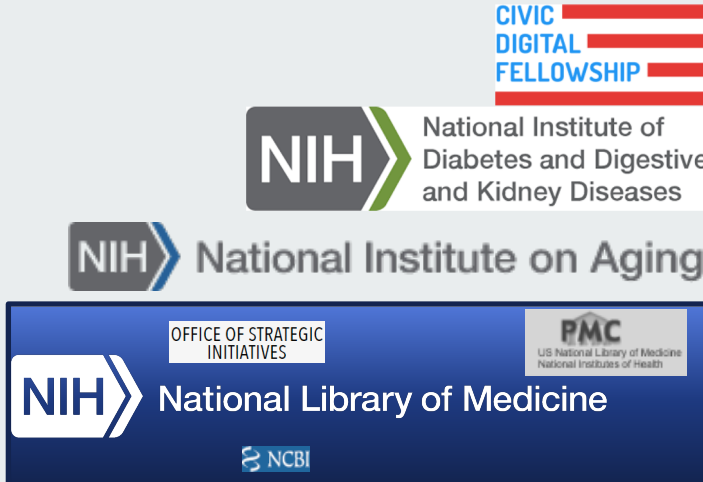
Data Sharing in PubMed Central

Presentation for the Civic Digital Fellowship Demo Day

Fellow: Natalie Gable
Mentors: Ken Wilkins, Jennie Larkin, Lisa Federer, Katie Funk, Rebecca Orris



Context

- **What we're working towards**
  - FAIR (findable, accessible, interoperable, and reusable) data principles underlie good sharing practices for papers published in PubMed Central (PMC)
- **Why is good data stewardship important?**
  - Reusability of data in academic studies
  - Reproducibility of results
  - Data citations and recognition for providing useful data resources
- **What currently exists (or what existed before the start of the project)?**
  - PubMed Central has features to tag Data Availability Statements (DASs; but it applies no strict enforcement of rules around submitting DASs)
  - Studies and analyses of data sharing in PLoS ONE (Federer *et al*) and Europe PMC (Parkin *et al*)

explore_pmc

Three parts to this project:

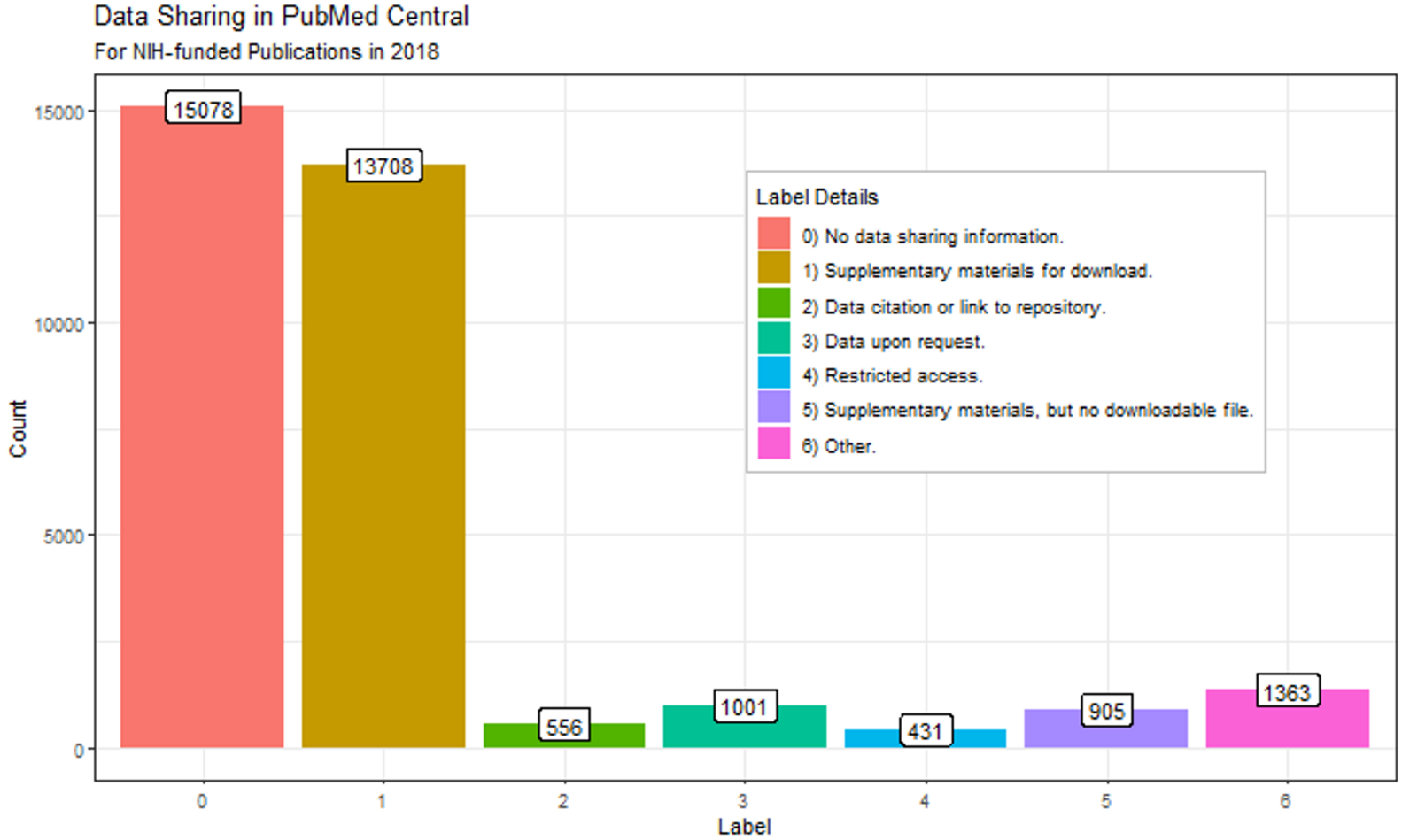
1) Build the infrastructure to handle data pulling/interfacing with PMC API in **R**
2) Curate a dataset and write functions to handle data embedding, preprocessing
3) Training, tuning, and testing models

Goal: write a package in **R** to interface with PubMed Central API and handle and model data; write specific functions to classify data sharing

The result: **explore_pmc**, a new suite of **R** functions and tools to search PubMed Central, pull text data from PubMed Central, and preprocess these text data for transformation into 1 or more 'embeddings' **(**quantitative learned representations).

Also a "**model zoo**", a collection of markdown files to show how to implement different classification models and some pretrained models with results.

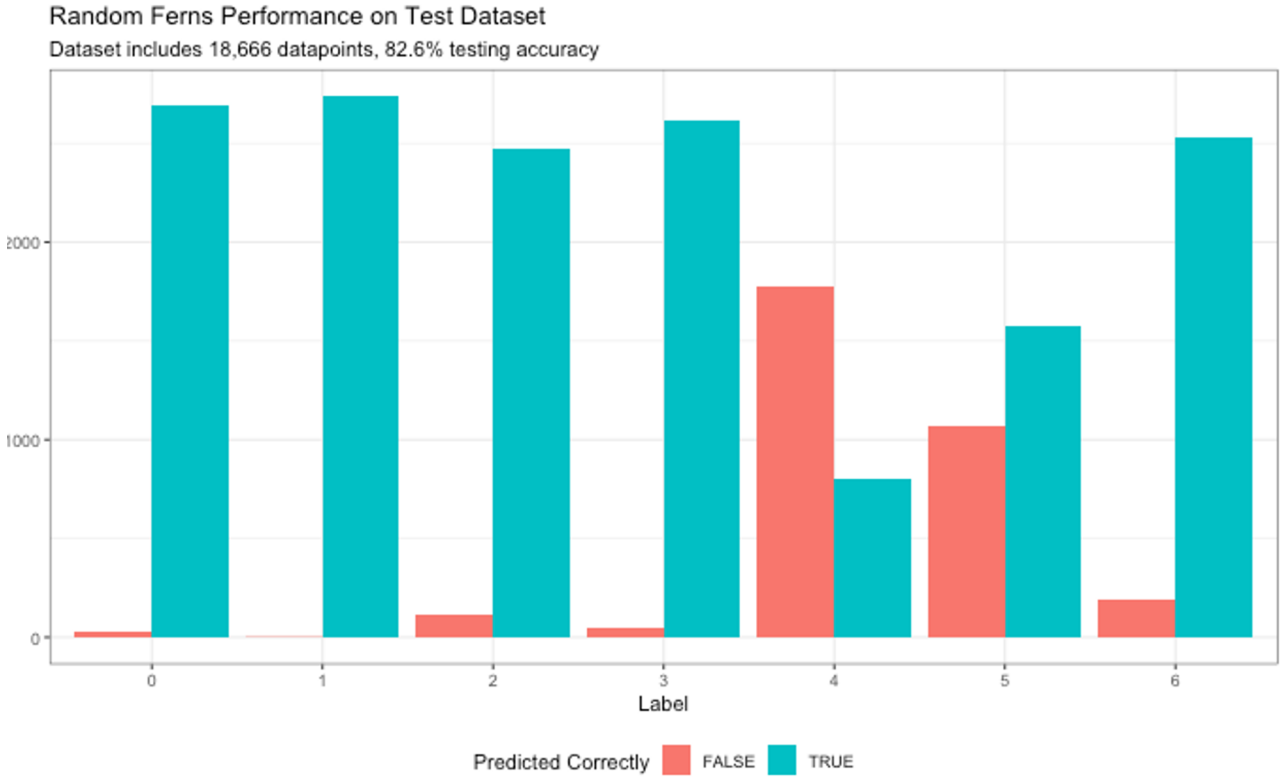
Data sharing in PMC



Models and Model Results

Tried out a few different models:

- K-nearest neighbors, random forest, SVM (with linear, polynomial, and radial kernels), multinomial regression, model averaged neural net
- Random ferns with latent Dirichlet allocation features had the best performance: 99% training accuracy and 83% testing accuracy



Future Directions: improve prediction of categories 4-6, leverage articles' MeSH terms